

Application for United States Letters Patent

for

METHOD AND APPARATUS FOR ADAPTIVE FRAME TRACKING

by

**Daniel B. Reents
Patrick Maupin**

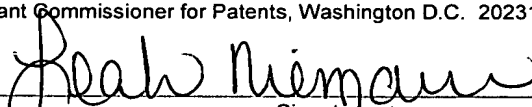
00375420-081699

EXPRESS MAIL MAILING LABEL

NUMBER EL332806391US

DATE OF DEPOSIT August 16, 1999

I hereby certify that this paper or fee is being deposited with the United States Postal Service "EXPRESS MAIL POST OFFICE TO ADDRESSEE" service under 37 C.F.R. 1.10 on the date indicated above and is addressed to:
Assistant Commissioner for Patents, Washington D.C. 20231.


Signature

METHOD AND APPARATUS FOR ADAPTIVE FRAME TRACKING

BACKGROUND OF THE INVENTION

1. FIELD OF THE INVENTION

5 This invention relates generally to data synchronization relating to telecommunications, and more particularly, to tracking frames of data during data transfers.

2. DESCRIPTION OF THE RELATED ART

Advancements in computer technology have facilitated developments of new methods
of computer system communications. One area of advancement in computer technology is
the improvement made in data bus communications. Recently, a new system of bus
communications, the Universal Serial Bus (USB), has improved data communications within
computer systems. Generally, to implement USB communications within a computer system,
a computer system must have an operating system that is designed to decipher USB-type
data. More recently, many computer systems that are produced contain software and
hardware that facilitate USB communications.

USB communications can also facilitate "plug and play" implementation. In other
words, using USB communications allows for computer peripherals to be implemented into
computer systems with ease. Computer peripherals such as scanners, digital joysticks, digital
speakers, or PC telephones can be attached to a computer system without the tedious task of
determining proper input ports and interrupts. Implementing USB into a computer system
can result in the capability of adding computer peripherals without the complications of
determining the complex protocols that are generally required for implementation of periph-
erals. In other words, adding a new digital scanner onto a computer system now does not

require adding a PC interface card, installing a software package, or changing computer interrupts; instead, the USB now allows for simple plug and play installation of the digital scanner.

5 Furthermore, USB allows for the capability of uninterrupted installation of computer peripherals into a computer system. This feature is generally known as "hot-installation." In fact, the USB permits the installation of electronic devices, such as generic computer peripherals, into computer systems without having to interrupt or shut down a task that is being conducted by a computer system. By implementing a "daisy-chain" system of installing
10 computer peripherals, USB allows for installation of multiple computer peripherals into a relatively small amount of USB input ports.

15 Another advantage of implementing USB systems in computer systems is the simplification of distribution of power for electronic devices that are interfaced with the computer system. USB ports are generally capable of supplying necessary electrical power to attached computer peripherals. This advantage reduces the need for independent power supplies for each electronic device that is interfaced with a computer system, thereby saving electrical power and valuable electronic real estate.

20 Generally, USB systems allow bi-directional data communications through its ports. In other words, a computer system can perform two-way communications with an electronic device that is interfaced with the computer system. The bi-directional communication feature that the USB provides can result in new and innovative methods of controlling computer peripherals. For example, by employing the USB system, a computer system can be utilized
25 to automatically manage a telephone communications center to maintain voice data, facsimile

data, and electronic data mailboxes. Also, utilizing the bi-directional communications system facilitated by USB systems facilitates computer system control of USB-compliant stereo speakers to match the acoustics of a particular listening environment.

5 USB system communications include sending and receiving packets of data to and from computer peripherals and computer systems. Many times, errors can occur when packets of data are transmitted and received. Often, packets of data, particularly isochronous frames of data, can lose their synchronous flow, resulting in transmission errors. The state of the art lacks an efficient method and means of recovering from data transmission errors. Furthermore, the state of the art does not currently provide an efficient means of automati-
10 cally compensating for data rate errors.

The present invention is directed to overcoming, or at least reducing the effects of, one or more of the problems set forth above.

SUMMARY OF THE INVENTION

15 In one aspect of the present invention, an apparatus is provided for performing adaptive frame tracking. The present invention comprises an adaptive frame tracking unit capable of receiving and sending at least one data packet and automatically adjusting a data rate of the data packet by determining if there exists at least one data frame error and correcting for the
20 data frame error in response to a determination that there exists at least one the data frame error.

25 In another aspect of the present invention, a method is provided for performing adaptive frame tracking. At least one data packet from a data stream is monitored. Transmit data

buffering of the data packet is supported. Whether at least one data packet is missing from a received data stream, is determined. A start of frame rate control for the data packet for the missing data packet, is established. An auto rate control of the data packet is supported.

5

BRIEF DESCRIPTION OF THE DRAWINGS

The invention may be understood by reference to the following description taken in conjunction with the accompanying drawings, in which like reference numerals identify like elements, and in which:

10 Figure 1 illustrates one embodiment of the apparatus, an adaptive frame tracking unit, of the present invention;

Figure 2 illustrates a flowchart depiction of the primary tasks performed by the method and apparatus of the present invention;

15 Figure 3 illustrates a more detailed block diagram depiction of the multiple sections of the frame tracking unit described in Figure 1;

20 Figure 4 illustrates one embodiment of the start of frame synthesizer circuit described in Figure 3;

Figure 5 illustrates one embodiment of the Universal Serial Bus frame position monitor circuit described in Figure 3;

Figure 6 illustrates one embodiment of the auto data rate control circuit described in Figure 3; and

Figure 7 illustrates one embodiment of the frame number monitor circuit described in Figure 3.

While the invention is susceptible to various modifications and alternative forms, specific embodiments thereof have been shown by way of example in the drawings and are herein described in detail. It should be understood, however, that the description herein of specific embodiments is not intended to limit the invention to the particular forms disclosed, but on the contrary, the intention is to cover all modifications, equivalents, and alternatives falling within the spirit and scope of the invention as defined by the appended claims.

DETAILED DESCRIPTION OF SPECIFIC EMBODIMENTS

Illustrative embodiments of the invention are described below. In the interest of clarity, not all features of an actual implementation are described in this specification. It will of course be appreciated that in the development of any such actual embodiment, numerous implementation-specific decisions must be made to achieve the developers' specific goals, such as compliance with system-related and business-related constraints, which will vary from one implementation to another. Moreover, it will be appreciated that such a development effort might be complex and time-consuming, but would nevertheless be a routine undertaking for those of ordinary skill in the art having the benefit of this disclosure.

The use of Universal Serial Bus (USB) systems in the field of computer technology has improved computer usage and computer communications. Implementing USB-related

solutions for computer technology can result in improvements in the quality and efficiency of computer system usage. The present invention provides for a method and apparatus for efficiently recovering from data packet framing errors, particularly start of frame errors. The present invention also provides for a computer software solution for determining a frame number of the data packet being transmitted on a USB system. Furthermore, the present invention provides for an automated method of adjusting a data rate of isochronous data transfers. The methods described by the present invention may be implemented through software, hardware, or firmware technologies. The methods described by the present invention may also be implemented by any combination of software, hardware, and firmware technologies.

Generally, packets of data are transmitted to and from a computer system and computer peripherals. The packets of data being transmitted generally contain start of frame sections, which contains information regarding the data packets being transmitted. In one embodiment, a USB system device, such as a USB host (not shown), transmits a set of generic data patterns along with specific, unique data. In one embodiment, the USB host is a computer system. The set of generic data patterns include uniform data patterns that aid computer systems in identifying a data packet, synchronizing the reception of data, and identifying the end of a particular data pattern. In one embodiment, a packet of data that is transmitted via a USB communication system includes a packet identification data pattern within the packet of data. In one embodiment, the packet identification data pattern is a start of frame (SOF) data pattern. Generally, a start of frame is sent by a USB host at a rate of once per frame of USB data.

Turning now to Figure 1, one embodiment of the apparatus, an adaptive frame tracking unit 110, of the present invention is illustrated. In one embodiment, digital logic circuits that are encompassed in the adaptive frame tracking unit 110 to operate in sync with the USB bit time clock on a line 120. A system reset signal on a line 130 is capable of resetting the logic functions of the digital logic circuits that are encompassed in the adaptive frame tracking unit 110. The adaptive frame tracking unit 110 receives a USB start of frame signal on a line 140 and performs digital functions to produce several digital signals for performing adaptive frame tracking. The adaptive frame tracking unit 110 also receives a frame number signal on a line 145. The digital signals generated by the adaptive frame tracking unit 110 includes the frame position updated signal on a line 150, a latched frame count signal on line 160, a frame position monitor sample clock on a line 165, an isochronous packet size signal on a line 170, a latched frame number signal on a line 180, and a time stamp match signal 190. In one embodiment, the steps comprising adaptive frame tracking is illustrated in Figure 2.

In one embodiment, the method and apparatus of the present invention monitors a USB start of frame data pattern. The adaptive frame tracking unit 110 is employed to support multiple data synchronization functions required by the USB system while it is transmitting or receiving data via the USB. In one embodiment, the start of frame packet sent by a host USB includes a sync data field, a packet identification data field, a frame number field, a CRC5 bit, and an end of packet (EOP) data field. In one embodiment, the frame number data field comprises of a twelve-bit data field. In one embodiment, the function of monitoring the start of frame data packet is important in supporting isochronous data transfers.

5 Isochronous data generally refers to time-dependant data. In other words, isochronous data transfer refers to processes where data is transmitted and received within certain time constraints. In one example, for multimedia data communications, an isochronous transport mechanism is generally required since data must be delivered as fast as it is displayed such that audio and video signals are synchronized. In contrast, asynchronous data transmission generally refers to processes in which random intervals can exist between transmission of packets of data. In further contrast with isochronous data transmission, synchronous data transmission can generally only be received or sent during specific intervals. Isochronous data transmission generally does not require a rigid protocol such as the one that is generally required for transmission of synchronous data. However, isochronous data transmission generally requires a more rigid protocol than the one required by the transmission of asynchronous data.

15 Turning now to Figure 2, a block diagram depiction of the primary tasks performed by the method and apparatus of the present invention, is illustrated. One of the primary functions of the adaptive frame tracking unit 110 is to monitor the USB start of frame data pattern, as described in block 210 of Figure 2. The adaptive frame tracking unit 110 supports a transmit data buffering function, as described in block 220 of Figure 2. Buffering the transmitted data allows the adaptive frame tracking unit 110 to control the USB data flow rate. The adaptive frame tracking unit 110 also determines whether at least one isochronous data packet was missed, as described in block 230 of Figure 2. The adaptive frame tracking unit 110 also establishes a data rate for the start of frame, as described in block 240 of Figure 2. The adaptive frame tracking unit 110 is capable of providing an automated data rate control for data transmission on a USB system, as described in block 250 of Figure 2.

The adaptive frame tracking unit 110 is capable of monitoring an USB system for a start of frame data packet and generating a start of frame data packet if one is not received. The start of frame data packet may be missing during data transmission for various reasons including the reasons that the USB host may not send a start of frame data packet or data on the USB has been corrupted. The creation of a start of frame data packet is generally known as start of frame synthesis. Turning now to Figure 3, more detailed block diagram depiction of the adaptive frame tracking unit 110, is illustrated. In one embodiment, the adaptive frame tracking unit 110 includes a start of frame synthesizer circuit 310, a frame position monitor circuit 320, a frame number monitor circuit 330, and an auto data rate control circuit 340.

The start of frame synthesizer circuit 310 generates a frame count signal on a line 350, which is sent to the frame position monitor circuit 320. The frame position monitor circuit 320 generates a frame position updated signal on the line 150, which is used to latch the frame count signal and create the latched frame count signal on the line 160. The frame position monitor 320 also generates a frame position monitor sample clock on a line 165. The start of frame synthesizer circuit 310 also generates a synthesized start of frame signal on a line 360, which are sent to the frame number monitor circuit 330 and the auto data rate control circuit 340. The auto data rate control circuit 340 generates an isochronous packet size signal on the line 170. The start of frame synthesizer circuit 310 also generates a missed start of frame signal on a line 370, which is sent to the frame number monitor circuit 330. The frame number monitor circuit 330 generates the latched frame number signal on the line 180, and the time stamp match signal on the line 190. The signals generated by the adaptive frame tracking unit 110 can be used in combination with a USB host to perform start of frame synthesis, start of frame monitoring, automatic data rate control, and other adaptive frame tracking tasks.

Turning now to Figure 4, one embodiment of a more detailed depiction of the start of frame synthesizer circuit 310, is illustrated. The start of frame synthesizer circuit 310 employs an adaptive frame tracking method to monitor data streams on the USB system to check for start of frame data patterns. The start of frame synthesizer circuit 310 is capable of detecting a missing start of frame data pattern. In one embodiment, the start of frame synthesizer circuit 310 then synthesizes a new start of frame data pattern and adds it to the data stream. The start of frame synthesis, in one embodiment, causes the start of frame interval length to be approximately the same as the start of frame interval length of the previous successfully received start of frame data pattern.

The start of frame synthesizer circuit 310 employs two counters, the frame counter 410 and the error counter 420, to track the number of bits in a particular frame of data. In one embodiment, the frame counter is incremented by the USB bit time clock on the line 120. The frame counter 410 is reset each time a start of frame is detected or each time a start of frame is synthesized. The output of the frame counter 410 is used to generate the frame count signal on a 350, which is sent to the frame position monitor circuit 320.

The error counter 420 is incremented or decremented based upon the number of bits in the data frame. The incrementing and the decrementing of the error counter 420 allow the synthesized start of frame length to be adaptable to the actual start of frame length. The outputs of the frame counter 410 and the error counter 420 are used to generate the synthesized start of frame signal on a line 360, which is sent to the auto data rate control circuit 340 and the frame number monitor circuit 330. In one embodiment, the synthesized start of frame

signal on the line 360 is generated at approximately the same equivalent period of time as the previous start of frame data packet.

The synthesized start of frame signal on the line 360 is generated earlier than the equivalent period of time as the previous start of frame data packet when the start of frame data packet is received before the expected time. Furthermore, the frame count signal on the line 350 is decremented when the start of frame data packet is received before the expected time period. The synthesized start of frame signal on the line 360 is generated at approximately the period of time as the previous start of frame data packet when the start of frame data packet is received later than the expected time period. The frame count signal on the line 350 is incremented when the start of frame data packet is received later than the expected time. The frame count signal on the line 350 is not incremented or decremented when the start of frame data packet is received at the expected time period. In one embodiment, an additional missed start of frame signal on the line 370 is generated when the start of frame data packet is not received within a predetermined period of time period after the start of frame data packet was expected.

Generally, synchronized isochronous data endpoints synchronize their data to approximately match the data rate of the USB host. In one embodiment, an isochronous data endpoint (endpoint) is a computer peripheral device. The adaptive frame tracking unit 110 is capable of monitoring USB start of frame data packet rate and modify the start of frame data packet rate to synchronize to the data rate of the USB host. In one embodiment, an external output that reflects the internally synthesized start of frame signal, is provided. The signal allows an external computer peripheral device, such as a digital signal processor (DSP), to monitor the data rate of the USB start of frame data packet. When the data rate of the USB

host start of frame data changes, the data rate of the isochronous endpoint is changed to track or adapt to the data rate of the USB host start of frame data. The adaptation of the data rate of the isochronous to the data rate of the USB host start of frame data is generally called adaptive synchronization.

5

In another embodiment, the adaptive frame tracking unit 110 is capable of synchronizing its data rate to a USB bit time clock on the line 120. This synchronization is performed by the adaptive frame tracking unit 110 by monitoring the data rate of USB data frames and requesting the USB host to change its data rate accordingly. In other words, the adaptive frame tracking unit 110 requests the USB host to change the number of USB data clocks in a single USB frame of data, thereby changing the USB data frame interval. The adaptation of the USB data frame interval is generally known as start of frame mastership. The frame position monitor circuit 320 is provided to allow the adaptive frame tracking unit 110 to perform the start of frame mastership.

Turning now to Figure 5, a more detailed embodiment of the frame position monitor circuit 320 is illustrated. In one embodiment, the frame position monitor circuit 320 monitors two different input clocks, and selects one clock to be a sample clock signal on a line 510. The frame monitor position circuit 320 detects the positive edge of the sample clock signal on a line 510, synchronizes the sample clock signal, and generates a single clock-wide frame position monitor sample clock on the line 165. Utilizing a dividing counter 540, the frame position monitor circuit 320 is capable of dividing the frame position monitor sample clock on the line 165 by 1, 2, 4, 8, 16, 32, or 64 times. The dividing factor used to divide the frame monitor sample clock on the line 165 is selected by the sample clock rate select signal on a line 550. The frame position monitor circuit generates a frame position updated signal on a

line 150 and utilizes this signal to latch the frame count signal on the line 350 to generate the latched frame count signal on the line 160. The frame count signal on the line 350 is latched into the frame count register 360. The latched frame count signal on the line 160 is latched on the positive edge of the divided frame position sample clock.

5

The latched frame count signal on the line 160 represents the data rate from a source, such as a computer peripheral plugged into a USB port of a computer system. The latched frame count signal can be read by the central process unit (CPU) (not shown) of a USB host. Software in a computer peripheral device can determine how its data rate is changing relative to a USB data clock by examining the latched frame count signal on the line 160 at various programmable time periods of the sample clock on the line 510 and request to the USB host to modify the USB data rate.

Turning now to Figure 6, a more detailed embodiment of the auto data rate control circuit 340, is illustrated. Utilizing the auto data rate control circuit 340, the adaptive frame tracking unit 110 is capable of automatically controlling the data rate of an isochronous data endpoint by using the data sample rate of the source, such as a computer peripheral, and the USB data frame rate. This is one form of adaptive data synchronization called auto rate control. The auto rate controller circuit 340 uses the frame position monitor sample clock on the line 165 to increment an auto rate counter 610. The auto rate counter 610 is incremented and cleared by the auto rate controller 620. The auto rate counter 610 is clocked by the USB bit time clock signal on the line 120. When the auto rate control circuit 340 receives the synthesized start of frame signal on the line 360, the count value in the auto rate counter 610 is latched into an auto rate register 630. The count value from the auto rate counter 610 is latched into the auto rate register 630 by a count value latch signal on a line 635, which is

generated by the auto rate controller 620. The auto rate counter 610 is then cleared and re-enabled by the auto rate controller 620. The latched count value from the auto rate counter 610 is then multiplied by 1, 2, or 4 by a left-shifter multiplier 640 and presented to the data endpoints as the isochronous packet size signal on the line 170. The multiplication factor used by the left-shifter multiplier 640 is controlled by the left shifter select signal on a line 650.

The isochronous packet size signal on the line 170 is used to automatically select the number of bytes that the isochronous endpoint will return to the USB host during the next IN transaction from the endpoint. The left shifter multiplier 640 is provided to support data sources that have more than one byte of data per sample. In one embodiment, the mechanism described in Figure 3 is implemented under an assumption that when the synthesized start of frame signal on a line 360 is generated, data that is to be sent to the USB host from a particular IN endpoint resides in a memory section (not shown) of a USB system. A plurality of data management systems that are known to those skilled in the art, may be employed to manage data movement processes between the memory section and a data endpoint.

Turning now to Figure 7, a more detailed depiction of the frame number monitor circuit 330, is illustrated. A start of frame data packet generally contains a USB frame number value, which is placed on the USB frame number signal on the line 145. The USB frame number signal on the line 145 is passed to a peripheral device, from the USB host, in each start of frame data packet that is successfully received by the peripheral device. The USB frame number value contained in the start of frame data packet is latched each time the synthesized start of frame signal on the line 360 is asserted. The frame number monitor circuit 330 latches the USB frame number signal using a frame number monitor counter 710

and generates the latched USB frame number signal on a line 720. The USB frame number signal may be then read to determine the present frame number value.

In one embodiment, the USB frame number signal may not be latched if a start of frame data packet is not received by the computer peripheral device in a particular frame of data. When a start of frame data packet is missing from a frame of data, the start of frame synthesizer circuit 310 generates the missed start of frame signal on the line 370, which is sent to the frame number monitor circuit 330. When the missed start of frame signal on the line 370 is asserted, the frame number monitor counter 710 increments the previously latched USB Frame number signal on the line 710, by one.

The frame number monitor circuit 330 includes a time stamp match register 730 that can be loaded, through a time stamp match signal on a line 740, by the USB system software. The time stamp match register 730 is utilized to generate a time stamp match signal on a line 750 when the latched frame number signal on the line 720 is equal to or greater than the time stamp match value signal on the line 740. In one embodiment, the time stamp match value is loaded into the time stamp register through the time stamp match signal on the line 750. In one embodiment, the time stamp match signal on the line 750 will be generated until the time stamp match register 730 is written again. In one embodiment, generating the latched frame number signal on the line 720 and the time stamp match signal on the line 750, provides the USB system software information regarding the frame number of the corrupted or missing start of frame data packet. The principle taught by the present invention may be utilized for other type of data communication systems.

The particular embodiments disclosed above are illustrative only, as the invention may be modified and practiced in different but equivalent manners apparent to those skilled in the art having the benefit of the teachings herein. Furthermore, no limitations are intended to the details of construction or design herein shown, other than as described in the claims

5 below. It is therefore evident that the particular embodiments disclosed above may be altered or modified and all such variations are considered within the scope and spirit of the invention. Accordingly, the protection sought herein is as set forth in the claims below.

669T30" 02T52E60